|  |
| --- |
| **EXAMEN 3ª EVALUACIÓN**  **MÓDULO: PROGRAMACIÓN 1º DAM DIURNO** |

**EXAMEN B**

|  |  |  |
| --- | --- | --- |
| **NOMBRE ALUMNO/A:** | **NOTA** |  |

**INDICACIONES DEL EXAMEN:**

* El examen se realizará en los ordenadores de clase.
* El examen se entregará por el aula virtual del módulo de Programación de 1º.
* Cada ejercicio se realizará en un único fichero, salvo que tengas que realizar clases.
* Los ficheros que den errores de compilación no se calificarán.
* Los nombres de los ficheros deben tener tu nombre, seguido del número del ejercicio.
* La herramienta para realizar el examen será el Notepad++ ó Eclipse.
* Da nombres significativos a las variables y métodos (en su caso), para facilitar la comprensión del planteamiento.
* Se calificará positivamente un planteamiento estructurado del problema, que los programas tengan una estructura sencilla y optimicen el código fuente.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **DETALLE DE CORRECCIÓN** | | | | | |
| **EJERCICIO 1** | | | **EJERCICIO 3** | | |
| **Método separar** | **0’75** |  | **Fichero directo** | **1’25** |  |
| **Método completar** | **0’75** |  | **Listado por autor** | **1’00** |  |
|  |  |  | **Menú** | **0’25** |  |
|  |  |  | **Buscar autor** | **0’50** |  |
|  |  |  | **Añadir autor** | **0’75** |  |
| **TOTAL** | **1,50** |  | **TOTAL** | **4,00** |  |
| **EJERCICIO 2** | | | **EJERCICIO 4** | | |
| **Procesar ficheros** | **0’75** |  | **Añadir y eliminar** | **0’50** |  |
| **Grabar ficheros** | **0’75** |  | **Cola vacía** | **0’25** |  |
| **Leer ficheros** | **0,25** |  | **Buscar y ver datos a eliminar** | **1,00** |  |
| **Estructura repetitiva (File)** | **0’75** |  | **Menú bucle** | **0’50** |  |
| **TOTAL** | **2,25** |  | **TOTAL** | **2’25** |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. Realizar una clase en Java denominada **Cadena**, nos va a servir para guardar caracteres. La clase la vas a implementar con los siguientes métodos, **que debes crear tú**:

* Los atributos de la clase van a ser código, nombre, apellidos y nacionalidad. El campo código va a ser de tipo entero y el resto de los campos son de tipo String pero con el siguiente tamaño: nombre con 17 bytes, apellido con 24 bytes y nacionalidad con 15 bytes.
* Crea el método **void separar(Clase Cadena, String)**, le pasamos una cadena de texto con todos los datos de los campos y debemos separarlos y guardarlos en las variables correspondientes de esta clase.
* Crear el método **String completar(String, int)**, le pasamos una cadena de caracteres y deseamos completarla con espacios en blanco hasta el valor entero indicado. Si la cadena pasada fuera más larga la acortamos al tamaño indicado en el int.

Crear el programa principal con las llamadas a los distintos métodos de la clase.

(Puntuación 1’25)

1. Me dan la carpeta DATOS, que contiene una serie de ficheros con los que vamos a trabajar. Los ficheros están grabados en formato binario y contienen números enteros que hacen referencia a las temperaturas que hemos tomado de una provincia en concreto, durante un periodo de tiempo. El propio nombre del fichero nos indica a qué provincia hacemos referencia.

Nos piden obtener para cada fichero lo siguiente:

1. La temperatura mínima.
2. La temperatura máxima.
3. La temperatura media.
4. Crear un fichero donde se almacenará: el nombre de la provincia y las tres temperaturas obtenidas (mínima, máxima y media). Su extensión será **new**.
5. Realizar la lectura del fichero generado para comprobar que los datos han sido almacenados de forma correcta.

**No podemos guardar todos los datos del fichero en una tabla y no vamos a teclear nosotros los nombres de los ficheros.**

(Puntuación 2’25)

1. Nos dan los ficheros de texto, con las siguientes características:

Cada registro está en una línea.

Los campos están separados por punto y coma.

Y los campos de tipo texto están encerrados entre comillas.

AUTOR.TXT

Que contiene un número entero que es el código\_autor, nombre, apellido y nacionalidad. Veamos un ejemplo:

15;"GERRALD";"DURREL";"BRITANICA"

16;"PEARL S.";"BUCK";"ESTADOUNIDENSE"

1;"C. S.";"Lewis";"IrlandESA"

LIBRO.TXT

Que contiene los datos en este orden, número de registro (entero), título del libro (cadena), editorial (cadena), precio (real), existencias (entero) y código\_autor (entero).

10;"Lo eterno sin disimulo";"Religioso";"RIALP";9,87;3;1

11;"El manuscrito encontrado en accra";"Religioso";"Planeta";15,71;21;2

51;"Harry Potter y el misterio del prín";"NOVELA";"Salamandra";10,50;1;5

Nos piden:

* + 1. Pasar el fichero AUTOR.TXT a un **fichero directo** según el código de autor. Para ello podemos utilizar el ejercicio 1.

**Recuerda que cada campo UTF que utilices emplea dos bytes más, de los caracteres que guarde.**

* + 1. Realizar un menú con lo siguiente que se os pide:
       1. Buscar un autor de forma directa.
       2. Añadir un nuevo escritor.
       3. Listado del fichero LIBRO según un autor pedido.

(Puntuación 4)

1. Nos piden realizar un programa para trabajar con una cola. El nodo de la cola va a tener un nombre y un dni. Debemos tener un programa que nos de las siguientes opciones:
   1. Añadir un elemento a la cola.
   2. Eliminar un elemento a la cola.
   3. Ver si la cola esta vacía.
   4. Buscar un elemento en la cola e indicar, cuantos elementos tendremos que borrar para llegar a él.

Debe ejecutarse tantas veces como queramos.

(Puntuación 2,25)